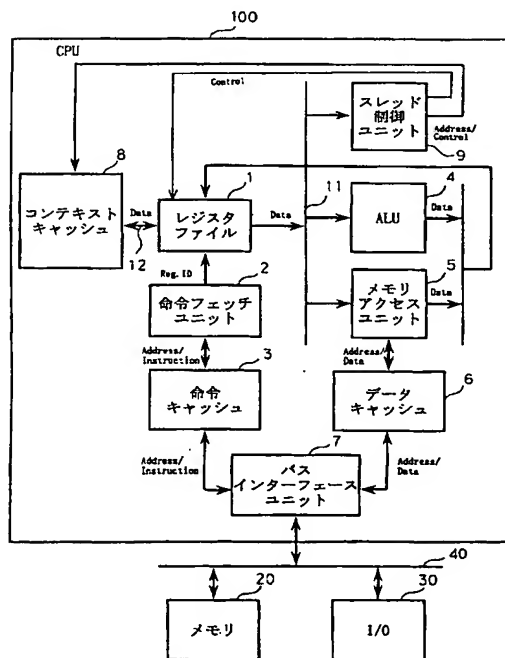




(10) 国際公開番号
WO 2004/063925 A1

- 添付公開書類：
一 国際調査報告書

(57) 要約: リアルタイムOSなどコンテキストの切り替えが頻繁に発生するアプリケーションにおいて、コンテキストの切り替えによるオーバーヘッドを大幅に削減する。OSは、Swap命令を発行し、コンテキストスイッチが開始する。Swap命令は、入れ替えるスレッドIDと共にスレッド制御ユニット9に発行される。スレッドIDは、コンテキストキャッシュ8に格納されているスレッドを一意に識別するために用いる。スレッド制御ユニット9は、コンテキスト専用バス12を経て、レジスタファイル1からコンテキストキャッシュ8へデータを退避すると同時に、コンテキストキャッシュ8からレジスタファイル1へ新しいスレッドのデータを送る。スレッド制御ユニット9は、送られてきたスレッドIDに基づいて自動的に必要な数だけレジスタファイル1内のデータとコンテキストキャッシュ8内のデータを入れ替える。



8...CONTEXT CACHE
1...REGISTER FILE
2...INSTRUCTION FETCH UNIT
3...INSTRUCTION CACHE
9...THREAD CONTROL UNIT
6...MEMORY ACCESS UNIT
5...DATA CACHE
7...BUS INTERFACE UNIT
20...MEMORY

明 細 書

コンテキスト切り替え方法・装置・プログラム・記憶媒体、中央演算装置

5

技術分野

本発明は、コンテキスト切り替え方法及び装置、中央演算装置、コンテキスト切り替えプログラム及びそれを記憶したコンピュータ読み取り可能な記憶媒体に係り、特に、リアルタイム・オペレーティングシステム(RT-OS)等のOSにおいてコン

10 テキストの切り替えによるオーバーヘッドの削減を可能にするコンテキスト切り替え方法及び装置、中央演算装置、コンテキスト切り替えプログラム及びそれを記憶したコンピュータ読み取り可能な記憶媒体に関する。ここで、コンテキストとは、例えば、汎用レジスタ、浮動小数点レジスタ、プログラムカウンタ、ステータスレジスタ等、記憶部(例、レジスタファイル)に記憶されている各スレッドの実行の

15 ための情報又は現在実行中の状態のことをいう。

背景技術

図8に、コンテキスト切り替え動作の説明図を示す。

この図では、複数のコンテキスト(スレッド)が1つの中央演算装置で切り替えられて実行されている例を示している。コンテキストが切り替わる場合、今まで実行していたコンテキストの状態(汎用レジスタ、浮動小数点レジスタ、プログラムカウンタ、ステータスレジスタなどであり、以下、単にコンテキストという場合がある。)を保存し、新しく実行するコンテキストの状態を読み出す必要がある。切り替え動作の際に要する時間をオーバーヘッドといい、オーバーヘッドは、コンテキストが

20

25 切り替わる度に発生する。

従来、コンテキスト切り替え時間を短縮するための技術としては、以下の文献が挙げられる。

特許文献1(特開平07-141208号)には、リアルタイムオペレーティングシス

テムを用いたマルチタスク処理装置において、各タスクに対応して占有される複数のレジスタバンクを設け、コンテキスト等の退避／復帰がレジスタバンクを切り替えることにより行い、ディスパッチ時間を短縮する技術が記載されている。

また、特許文献2(特開平09-212371号)には、マルチタスク処理を行うマイクロプロセッサにおいて、複数のレジスタに対応してその内容の変化又は非変化を示すビットを設け、タスク切り替えが発生したときに、そのビットに従い、レジスタの内容が変化したときに退避命令を実行し、変化しないとき退避命令を実行しないようにすることで、OSのオーバーヘッドを減少させるレジスタ退避及び復元システムが記載されている。

10

発明の開示

従来の方法では、コンテキストを切り替える場合、中央演算装置内に保持されているコンテキストの状態は、OS等のソフトウェアによりストア命令を用いて1つずつ中央演算装置外の記憶装置に保存される。その後、OS等のソフトウェアは、ロード命令を用いて新しいコンテキストを記憶装置から読み込む。つまり、コンテキストを切り替える度に、コンテキストの保存と読み込みのためのメモリアクセスが数百サイクルから千数百サイクルも生じることになり、大きなオーバーヘッドとなっている。また、従来の方法では、ロード命令やストア命令を用いたソフトウェアによるコンテキストの保存と読み込みを行っているため、1度に1つのデータしか扱うことができない。そのため保存すべき状態が増えるとコンテキストスイッチにかかる時間も増加する。

本発明は、以上の点に鑑み、特に、リアルタイムOSなどコンテキストの切り替えが頻繁に発生するアプリケーションにおいて、コンテキストの切り替え(コンテキストスイッチ)によるオーバーヘッドを大幅に削減することを目的とする。本発明は、例えば、コンテキストを切り替える度に、コンテキストの保存と読み込みのためのメモリアクセスが、1サイクルから数サイクルで可能とするコンテキスト切り替え方法及び装置、中央演算装置、コンテキスト切り替えプログラム及びそれを記憶したコンピュータ読み取り可能な記憶媒体を提供することを目的とする。

また、本発明は、特に実時間処理システムのようにコンテキストスイッチが頻繁

に起こるシステムにおいて、コンテキストスイッチにかかる時間を一定とし、実時間性の時間粒度を非常に小さくすることを目的とする。

本発明は、特に、

1. コンテキストを保持するための専用記憶装置(コンテキストキャッシュ)を持ち、

2. レジスタよりビット幅の広い専用バスを用いて専用記憶装置(コンテキストキャッシュ)と中央演算装置(CPU)を接続する、

ことにより、コンテキスト切り替えにかかるオーバーヘッドを削減する。

本発明の第1の解決手段によると、

10 複数のコンテキストを切り替えるコンテキスト切り替え装置であって、
演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関するコンテキストが記憶されたレジスタファイルと、

前記レジスタファイルに接続され、コンテキストをキャッシュするためのコンテキストキャッシュと、

15 前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキストスイッチ専用バスと、

前記コンテキストキャッシュに記憶されているスレッドのコンテキストを識別するためのスレッド識別子(スレッドID)を記憶するスレッドIDテーブルを有し、演算論理ユニット及びメモリアクセスユニットと並列に接続され、前記コンテキストキャッシュと前記レジスタファイルとの間のコンテキストのデータ転送を制御するスレッド制御ユニットと

を備え、

前記スレッド制御ユニットは、

25 コンテキストの切り替えが発生した場合、入力された切り替え命令と新たに入れ替えるスレッド識別子(スレッドID)に基付き前記スレッドIDテーブルを検索し、

新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッシュのアドレスと、実行中のコンテキストが記憶されている前記レジスタファイルのレジスタ識別子を求め、

求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求めた

レジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジスタファイル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテキストスイッチ専用バスを介して入れ替え、退避又は復帰を実行する

前記コンテキスト切り替え装置

5 が提供される。

本発明の第2の解決手段によると、

上述のようなコンテキスト切り替え装置と、

命令及びデータをそれぞれキャッシュする命令キャッシュ及びデータキャッシュと、

10 前記命令キャッシュから命令をフェッチ及びデコードする命令フェッチユニットと、
レジスタファイルに記憶された命令に従い各種演算を行い、演算結果を前記レジスタファイルに書き戻す演算論理ユニットと、

前記レジスタファイルからオペランドと命令が送られ、前記データキャッシュをアクセスし、ロード又はストアを実行するメモリアクセスユニットと、

15 前記レジスタファイル、前記演算論理ユニット、前記メモリアクセスユニット及び前記スレッド制御ユニットを並列に接続する演算バスと、

を備えた中央演算装置

が提供される。

本発明の第3の解決手段によると、

20 演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関するコンテキストが記憶されたレジスタファイルと、

前記レジスタファイルに接続され、コンテキストをキャッシュするためのコンテキストキャッシュと、

前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキストスイッチ専用バスと
25

を備えたコンテキスト切り替え装置を用い、複数のコンテキストを切り替えるコンテキスト切り替え方法であって、

コンテキストの切り替えが発生した場合、入力された切り替え命令と新たに入れ替えるスレッド識別子(スレッドID)に基付き、前記コンテキストキャッシュに記憶さ

れているスレッドのコンテキストを識別するためのスレッド識別子(スレッドID)を記憶する前記スレッドIDテーブルを検索し、

- 新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッシュのアドレスと、実行中のコンテキストが記憶されている前記レジスタファイルのレジスタ識別子を求め、

求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求めたレジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジスタファイル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテキストスイッチ専用バスを介して入れ替え、退避又は復帰を実行する

- 10 前記コンテキスト切り替え方法
が提供される。

さらに、本発明の第4の解決手段によると、

演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関するコンテキストが記憶されたレジスタファイルと、

- 15 前記レジスタファイルに接続され、コンテキストをキャッシュするためのコンテキストキャッシュと、

前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキストスイッチ専用バスと

- を備えたコンテキスト切り替え装置を用い、コンピュータが複数のコンテキストを切り替えるコンテキスト切り替えプログラムを記録したコンピュータ読み取り可能な記録媒体であって、

- コンテキストの切り替えが発生した場合、入力された切り替え命令と新たに入れ替えるスレッド識別子(スレッドID)に基付き、前記コンテキストキャッシュに格納されているスレッドのコンテキストを識別するためのスレッド識別子(スレッドID)を記憶する前記スレッドIDテーブルを検索するステップと、

新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッシュのアドレスと、実行中のコンテキストが記憶されている前記レジスタファイルのレジスタ識別子を求めるステップと、

求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求めた

- レジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジスタファイル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテキストスイッチ専用バスを介して入れ替え、退避又は復帰を実行するステップと
- 5 コンピュータに実行させるためのコンテキスト切り替えプログラム、及び、そのプログラムを記録したコンピュータ読み取り可能な記録媒体
- が提供される。

図面の簡単な説明

- 第1図は、一般的なCPUを含むコンピュータの構成図である。
- 10 第2図は、一般的なCPUでのコンテキストスイッチのフローチャートである。
- 第3図は、コンテキストキャッシュを用いたCPUを含むコンピュータの構成図である。
- 第4図は、コンテキストキャッシュを用いたCPUでのコンテキストスイッチのフローチャートである。
- 15 第5図は、コンテキスト切り替え装置の詳細構成図である。
- 第6図は、Swap命令のフローチャートである。
- 第7図は、コンテキスト切り替え装置の実装に関する説明図である。
- 第8図は、コンテキスト切り替え動作の説明図である。

20 発明を実施するための最良の形態

1. 関連技術

- なお、本実施の形態は、一例として、並列分散リアルタイム制御用 `Responsive Multi-Threaded (RMT) Processor` の中央演算装置上でコンテキストキャッシュとして利用されている。そこで、まず、本実施の形態が関連するCPUを有するコンピュータについて説明する。
- 25

図1に、一般的なCPUを含むコンピュータの構成図を示す。

このコンピュータは、CPU10、メモリ20、I/O30、バス40

を有する。CPU10は、バス40を介して、メモリ20、I/O30等と接続されている。CPU10は、レジスタファイル1、命令フェッチユニット2、命令キャッシュ3、演算論理ユニット(arithmetic logic unit、ALU)4、メモリアクセスユニット5、データ
5 キャッシュ6、バスインターフェースユニット7、演算バス11を備える。

レジスタファイル1は、汎用レジスタ(GPR)、浮動小数点レジスタ(FPR)、プログラムカウンタ(PC)、ステータスレジスタ(SR)等の各種レジスタを含む。レジスタファイル1は、現在実行中のコンテキストを記憶する。命令キャッシュ3及びデータ
10 キャッシュ6は、例えば、SRAM、フリップフロップ(FF)等の素子が用いられ、アクセス、読み出し、書き込み等の処理速度が速いものの記憶容量が小さい。一方、CPU10外部のメモリ20は、SDRAM、DRAM等の素子が用いられ、アクセス、読み出し、書き込み等の処理速度がキャッシュより遅いものの記憶容量が大きい。

ALU4は、局所性原理に従い、命令キャッシュ3、データキャッシュ6又はメモリ
15 20から必要なデータ及び命令を利用する。ALU4は、命令キャッシュ3又はデータキャッシュ6に処理に必要な命令又はデータがあればそれを利用し、なければメモリ20にアクセスして必要な命令又はデータを取得する。また、バスインターフェースユニット7は、命令キャッシュ3、メモリアクセスユニット5、データキャッシュ6とCPU外部のメモリ20、I/O30等をバス40を介して接続し、CPU内部と外部の間で
20 データの入出力を行うユニットである。演算バス11は、レジスタファイル1、ALU4、メモリアクセスユニット5を並列に接続する演算パイプライン等のバスである。

命令フェッチユニット2は、命令キャッシュ3へアドレス(Address)を出力し、命令キャッシュ3から命令(instruction)をフェッチし、デコードする。デコードされた命令に従い、ALU4がレジスタファイル1から必要なオペランドを読み出す。ALU
25 4は、それに従い各種演算を行い、演算結果をレジスタファイル1に書き戻す。ロード(Load)又はストア(Store)等のメモリアクセス命令の場合、デコードされた命令に従い、メモリアクセスユニット5がオペランドを読み出す。

メモリアクセスユニット5は、データキャッシュ6にアクセスしLoad又はStoreを行う。Store命令の場合、メモリアクセスユニット5は、アドレスとデータをデータキャ

ツシュ6に送り、データをデータキャッシュ6に格納する。Load命令の場合、メモリアクセスユニット5は、アドレスをデータキャッシュ6に送りデータキャッシュ6からデータを読み出す。読み出したデータは、レジスタファイル1に書き戻される。このとき、データキャッシュ6に求めるデータがなければ、メモリ20からそれを読み出す。

このような一般的な構成の場合、レジスタファイル1に記憶されたコンテキストの退避はStore命令を用いて行う。

図2に、一般的なCPUでのコンテキストスイッチのフローチャートを示す。

OS等のソフトウェアはStore命令を発行し、コンテキストスイッチを開始する。Store命令が発行されると、レジスタファイル1からデータが読み出され、メモリアクセスユニット5に送られる。そして、メモリアクセスユニット5は、データを格納するアドレスを計算し、データキャッシュ6にアクセスする(S101)。もし、データキャッシュミスが発生した場合(S103)、データキャッシュ6はメモリ20からキャッシュラインを読み込む(S105)。一方、データキャッシュミスが発生しなかった場合(S103)は、ステップS107へ進む。データキャッシュ6は、メモリアクセスユニット5から送られてきたデータを適当なアドレスで自身のキャッシュ内に格納する(S107)。OS等のソフトウェアは、このStore命令の処理を退避すべきレジスタの数だけ繰り返す。すなわち、全てのレジスタの内容の保存が完了していない場合は、ステップS101へ戻り処理を続け、全てのレジスタの内容の保存が完了した場合は、ステップS111へ進む(S109)。

つぎに、OS等のソフトウェアは、新しく実行されるコンテキストの復帰を、Load命令を用いて行う。実行中のコンテキストの保存が完了すると、OS等のソフトウェアは、Load命令を発行する。Load命令が発行されると、メモリアクセスユニット5は、データを読み込むためのアドレスを計算し、データキャッシュ6にアクセスする(S111)。もし、データキャッシュミスが発生した場合(S113)、データキャッシュ6はメモリ20からキャッシュラインを読み込む(S115)。データキャッシュミスが発生しなかった場合(S113)は、ステップS117へ進む。メモリアクセスユニット5は、データキャッシュ6からデータが返ってくる(S117)と、それをレジスタファイル1に書き戻す。このLoad命令を復帰すべきレジスタの数だけ繰り返す。すなわち、

全てのレジスタ内容の読み込みが完了した場合は、ステップS111へ戻り処理を続け、全てのレジスタ内容の読み込みが完了した場合は、コンテキストスイッチを終了する(S119)。

5 2. コンテキスト切り替え装置を備えたCPU

図3に、コンテキストキャッシュを用いたCPUを含むコンピュータの構成図を示す。

このCPU100は、レジスタファイル1、命令フェッチユニット2、命令キャッシュ3、ALU4、メモリアクセスユニット5、データキャッシュ6、バスインターフェースユニット7、コンテキストキャッシュ8、スレッド制御ユニット9、演算バス11、コンテキスト専用バス12を備える。演算バス11は、レジスタファイル1、ALU4、メモリアクセスユニット5にさらにスレッド制御ユニット9を並列に接続する演算パイプライン等のバスである。図1のCPU10の各部に対応して、同符号で示される各部の構成及び動作は、上述した通りである。

15 コンテキストキャッシュ8は、SRAM、FF等の素子が用いられ、アクセス、読み出し、書き込み等の処理速度が速い。コンテキストキャッシュ8は、レジスタファイル1にコンテキストスイッチ専用バス12により接続され、コンテキストをキャッシュするために用いられる。スレッド制御ユニット8は、コンテキストキャッシュ8を制御するためのユニットであり、ALU4やメモリアクセスユニット5と並列に接続されている。なお、スレッドとは、一般に、OSがあるプロセス又はタスクを並列処理するため、プロセス又はタスクを分割した処理単位又は最小単位のことをいう。この際、プロセス又はタスクが分割されることなく、1プロセス又は1タスクが1スレッドとなる場合もある。コンテキストスイッチが起こった場合には、現在実行されているスレッドのコンテキスト(汎用レジスタ、浮動小数点レジスタ、プログラムカウンタ、ステータスレジスタ等)を退避し、新しく実行されるスレッドのコンテキストを復帰する
20 必要がある。本実施の形態によるコンテキストキャッシュ8を用いたコンテキストスイッチの場合、コンテキストの退避と復帰は、レジスタファイル1とコンテキストキャッシュ8のデータをコンテキストスイッチ専用バス12を経て入れ替えるスワップ(Swap)命令を用いる。
25

図4に、コンテキストキャッシュを用いたCPUでのコンテキストスイッチのフローチャートを示す。

OS等のソフトウェアは、Swap命令を発行し、コンテキストスイッチが開始する。Swap命令は、入れ替えるスレッドIDと共にスレッド制御ユニット9に発行される

5 (S201)。スレッドIDは、コンテキストキャッシュ8に格納されているスレッドを一意に識別するために用いる。スレッド制御ユニット9は、コンテキストスイッチ専用バス12を経て、レジスタファイル1からコンテキストキャッシュ8へデータを退避すると同時並列に、コンテキストキャッシュ8からレジスタファイル1へ新しいスレッドのデータを送る。スレッド制御ユニット9は、送られてきたスレッドIDに基づいて自動

10 的に必要な数だけレジスタファイル1内のデータとコンテキストキャッシュ8内のデータを入れ替える(S203)。このように、OS等のソフトウェアはSwap命令を発行するだけで、専用ハードウェアがコンテキストスイッチを開始し終了する。

3. コンテキスト切り替え装置の詳細

15 図5に、コンテキスト切り替え装置の詳細構成図を示す。

レジスタファイル1は、汎用レジスタ111、浮動小数点レジスタ112、プログラムカウンタ113、ステータスレジスタ114を有する。コンテキストキャッシュ8は、コンテキストを格納するコンテキスト記憶領域8-1、8-2、...、8-nを所定のコン

20 テキスト数有する。スレッド制御ユニット9は、オンチップのコンテキストキャッシュ8とレジスタファイル1を制御するためのコントローラである。スレッド制御ユニット9内部にはオンチップメモリに格納されているコンテキストを識別するためのスレッドIDを所定数保持するためのスレッドIDテーブル91を設ける。また、マルチスレッドプロセッサの場合は、レジスタファイル1が並列に複数存在することになる。

図6に、Swap命令のフローチャートを示す。

25 コンテキストの切り替えは、例えばコンテキストスイッチハンドラー内でコンテキストスイッチ専用命令を用いることができる。OS等のソフトウェアが、Swap命令を発行すると、専用ハードウェアであるスレッド制御ユニット9が、そのSwap命令とスレッドIDを受けとる(S300)。スレッド制御ユニット9は、スレッドIDに基づいてスレッドIDテーブル91を検索し、コンテキストキャッシュ8をアクセスするため、入

れ替えるべきスレッドのデータ(コンテキスト)が格納されているアドレスを計算し、また、レジスタファイル1をアクセスするため、レジスタIDを計算する(S301)。つぎに、全てのコンテキストのデータ(汎用レジスタ、ステータスレジスタなど)に対して、ステップS302からS304により繰返しループ処理が行われる。スレッド制御

5 ユニット9は、計算したアドレスをもとにコンテキストキャッシュ8をアクセスし、新しく入れ替えるスレッドのデータ(コンテキスト)を読み出し、レジスタファイル1にそれを書き込む(S303)。それと同時又は並列に、スレッド制御ユニット9は、レジスタファイル1にアクセスし、今実行されているスレッドのデータ(コンテキスト)を読み出し、コンテキストキャッシュ8にそれを書き込む(S303)。このようにして、レ

10 ジスタファイル1とコンテキストキャッシュ8は、それぞれのデータを入れ替える。全てのデータの入れ替えが終了していない場合(S304)、スレッド制御ユニット9は、コンテキストキャッシュ8をアクセスするアドレスとレジスタファイル1をアクセスするレジスタIDを1つずつ増やし、ステップS303に戻って処理を続ける(ステップS302及びS304による繰返しループ処理)。全てのデータの入れ替えが終了した場合

15 (S304)、Swap命令を終了する。

このように、スレッド制御ユニット9は、コンテキストキャッシュ8とレジスタファイル1のそれぞれのアドレスを連続的にインクリメントすることにより、必要な数だけデータを入れ替える。この際、従来のようにLoad及びStore命令を用いた場合には数百～千数百クロックサイクルかかっていたコンテキストスイッチを、本実施の

20 形態では、レジスタファイル1とコンテキストキャッシュ8のコンテキストスイッチ専用バス12によるデータ転送幅をレジスタのビット幅に比べて非常に広くすることによって、1クロックサイクル～数クロックサイクル程度で実現する。具体的には、複数個のレジスタをまとめて1つの巨大なレジスタとして、その巨大なレジスタ毎に前記レジスタIDを割り当てるようにする。この際、全てのレジスタをまとめて1つ

25 の巨大なレジスタとした場合、1クロックサイクルでコンテキストスイッチを実現できる。

スレッド制御ユニット9は、コンテキストを入れ替えるSwap命令の他に、コンテキストを退避するバックアップ(Backup)命令、コンテキストを復帰するリストア(Restore)命令を処理する。Backup命令の場合は、コンテキストキャッシュ8か

らレジスタファイル1へのコンテキストのデータ転送は行われず、レジスタファイル1からコンテキストキャッシュ8へのコンテキストのデータ転送のみが実行される。一方、Restore命令の場合は逆にレジスタファイル1からコンテキストキャッシュ8へのコンテキストのデータ転送は行われず、コンテキストキャッシュ8からレジスタ

5 ファイル1へのコンテキストのデータ転送のみが実行される。

つぎに、図7に、コンテキスト切り替え装置の実装に関する説明図を示す。

コンテキストキャッシュ8は、コンテキストをバックアップするオンチップのメモリであり、この例では2ポートを有し、CPUにオンチップで実装されている。この例では、コンテキストキャッシュ8は、書き込みポート82、読み出しポート83を備える。コン

10 テキストキャッシュ8は、所定の複数コンテキスト分（例えば、32個等）の記憶領域を含むことができる。

レジスタファイル1は、通常のリードポートとライトポートの他にコンテキスト切り替え用の専用ポート、即ちコンテキストスイッチ用読み出しポート17とコンテキストスイッチ用書き込みポート18を付加し、これらのポートにコンテキストを保持するための記憶装置を接続する。この例では、レジスタファイル1は、レジスタ読み

15 出しポート15、レジスタ書き込みポート16、コンテキストスイッチ用読み出しポート17、コンテキストスイッチ用書き込みポート18を備える。レジスタ読み出しポート15は、レジスタファイル1からCPU内の装置へのレジスタの読み出しを、レジスタ書き込みポート16は、CPU内の装置からレジスタファイル1へのレジスタの書

20 き込みを、コンテキストスイッチ用読み出しポート17は、レジスタファイル1からコンテキストファイル8へのレジスタの読み出しを、コンテキストスイッチ用書き込みポート18は、コンテキストファイル8からレジスタファイル1へのレジスタの書き込みを、それぞれ行うポートである。

コンテキストの切り替えが発生した場合、OS等のソフトウェアはSwap命令を発

25 行し、スレッド制御ユニットが、これらのポート17、18を通して中央演算装置のコンテキストキャッシュ8内のデータをレジスタファイル1に保持し、新しいコンテキストをこのレジスタファイル1から取り出す。また、レジスタファイル1とコンテキストキャッシュ8を接続するコンテキスト専用バス12-1、12-2の幅をレジスタファイル1のビット幅より広くすることにより、1度に入れ替えるデータの量を増やす。こ

の例では、レジスタファイル1とオンチップメモリのコンテキストキャッシュ8とは、256ビットのコンテキストスイッチ専用バス12-1、12-2で繋がれる。また、コンテキストキャッシュ8として2ポートのオンチップメモリを用いることにより読み込みと書き出しを同時に行うことができる。そのため32ビット、32本の汎用レジスタは

5 4クロックサイクルでコンテキストを入れ替えることができる。

なお、ビット数、記憶容量、ポート数等の各種パラメータは一例に過ぎず、適宜設定することができる。

4. その他

10 本発明のコンテキスト切り替え方法又はコンテキスト切り替え装置・システムは、その各手順をコンピュータに実行させるためのコンテキスト切り替えプログラム、コンテキスト切り替えプログラムを記録したコンピュータ読み取り可能な記録媒体、コンテキスト切り替えプログラムを含みコンピュータの内部メモリにロード可能なプログラム製品、そのプログラムを含むサーバ等のコンピュータ、等により提供され

15 ることができる。

産業上の利用可能性

本発明によると、以上のように、特に、リアルタイムOSなどコンテキストの切り替えが頻繁に発生するアプリケーションにおいて、コンテキストの切り替え(コンテキストスイッチ)によるオーバーヘッドを大幅に削減することができる。また、本発明によると、例えば、コンテキストを切り替える度に、コンテキストの保存と読み込みのためのメモリアクセスが、1サイクルから数サイクルで可能とするコンテキスト切り替え方法及び装置、中央演算装置、コンテキスト切り替えプログラムを記録したコンピュータ読み取り可能な記録媒体を提供することができる。

20

25 また、本発明によると、特に、実時間処理システムのようにコンテキストスイッチが頻繁に起こるシステムにおいて、コンテキストスイッチにかかる時間を一定とし、実時間性の時間粒度を非常に小さくすることができる。

請 求 の 範 囲

1. 複数のコンテキストを切り替えるコンテキスト切り替え装置であって、
演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関する
コンテキストが記憶されたレジスタファイルと、
5 前記レジスタファイルに接続され、コンテキストをキャッシュするためのコンテ
キストキャッシュと、
前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキストス
イッチ専用バスと、
前記コンテキストキャッシュに記憶されているスレッドのコンテキストを識別す
10 るためのスレッド識別子(スレッドID)を記憶するスレッドIDテーブルを有し、演
算論理ユニット及びメモリアクセスユニットと並列に接続され、前記コンテキスト
キャッシュと前記レジスタファイルとの間のデータ伝送を制御するスレッド制御
ユニットと
を備え、
15 前記スレッド制御ユニットは、
コンテキストの切り替えが発生した場合、入力された切り替え命令と新たに入
れ替えるスレッド識別子(スレッドID)に基づき前記スレッドIDテーブルを検索
し、
新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッシュ
20 のアドレスと、実行中のコンテキストが記憶されている前記レジスタファイルの
レジスタ識別子を求め、
求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求め
たレジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジスタファ
イル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテキスト
25 スイッチ専用バスを介して入れ替え、退避又は復帰を実行する
前記コンテキスト切り替え装置。
2. 前記コンテキストスイッチ専用バスは、バス幅をレジスタ長以上とすることで、
1度に複数のコンテキストのデータを同時にスワップ、バックアップ又はリストア

する請求項1に記載のコンテキスト切り替え装置。

3. 前記コンテキストキャッシュは、リードポートとライトポートを備え、

5 前記レジスタファイルは、リードポートとライトポート、コンテキストスイッチ用
リードポートとライトポートを備え、

前記コンテキストキャッシュのリードポートとライトポートは、それぞれ前記コン
テキストスイッチ専用バスにより前記レジスタファイルのコンテキストスイッチ
用ライトポートとリードポートと接続されることを特徴とする請求項1に記載のコン
テキスト切り替え装置。

10

4. 前記スレッド制御ユニットは、コンテキストキャッシュにキャッシュされた所定
数のコンテキストを識別する所定数のスレッドIDテーブルを含むことを特徴とす
る請求項1に記載のコンテキスト切り替え装置。

15 5. 前記スレッド制御ユニットは、コンテキストを入れ替えるスワップ命令を実行
する場合、オペレーティングシステム等のソフトウェアからスレッド識別子(スレ
ッドID)をオペランドとして含むスワップ命令が発行されると、前記レジスタファ
イルから前記コンテキストキャッシュへ実行中のスレッドのコンテキストを退避
すると同時に、前記コンテキストキャッシュから前記レジスタファイルへ新しく入
20 れ替えるスレッドのコンテキストを送り、自動的に必要な数だけ前記レジスタフ
ァイル内のデータと前記コンテキストキャッシュ内のデータを入れ替えることを
特徴とする請求項1乃至4のいずれかに記載のコンテキスト切り替え装置。

25 6. 前記スレッド制御ユニットは、コンテキストを退避するバックアップ命令を実行
する場合、オペレーティングシステム等のソフトウェアからスレッド識別子(スレ
ッドID)をオペランドとして含むバックアップ命令が発行されると、前記レジスタ
ファイルから前記コンテキストキャッシュへのコンテキストのデータ転送を行い、
前記コンテキストキャッシュから前記レジスタファイルへのコンテキストのデータ
転送は行わないことを特徴とする請求項1乃至4のいずれかに記載のコンテキ

スト切り替え装置。

7. 前記スレッド制御ユニットは、コンテキストを復帰するリストア命令を処理する場合、オペレーティングシステム等のソフトウェアからスレッド識別子(スレッドID)をオペランドとして含むリストア命令が発行されると、前記コンテキストキャッシュから前記レジスタファイルへのコンテキストのデータ転送を行い、前記レジスタファイルから前記コンテキストキャッシュへのコンテキストのデータ転送は行わないことを特徴とする請求項1乃至4のいずれかに記載のコンテキスト切り替え装置。
8. 前記請求項1乃至7のいずれかに記載のコンテキスト切り替え装置と、
命令及びデータをそれぞれキャッシュする命令キャッシュ及びデータキャッシュと、
前記命令キャッシュから命令をフェッチ及びデコードする命令フェッチユニットと、
レジスタファイルに記憶された命令に従い各種演算を行い、演算結果を前記レジスタファイルに書き戻す演算論理ユニットと、
前記レジスタファイルからオペランドと命令が送られ、前記データキャッシュをアクセスし、ロード又はストアを実行するメモリアクセスユニットと、
前記レジスタファイル、前記演算論理ユニット、前記メモリアクセスユニット及び前記スレッド制御ユニットを並列に接続する演算バスと、
を備えた中央演算装置。
9. 前記メモリアクセスユニットは、ストア命令の場合はアドレスとデータを前記データキャッシュに送り、データを前記データキャッシュに格納し、ロード命令の場合はアドレスを前記データキャッシュに送り、データを前記データキャッシュから読み出し、読み出したデータを前記レジスタファイルに書き戻す請求項8に記載の中央演算装置。

10. 演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関する
コンテキストが記憶されたレジスタファイルと、

前記レジスタファイルに接続され、コンテキストをキャッシュするためのコン
テキストキャッシュと、

5 前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキスト
スイッチ専用バスと

を備えたコンテキスト切り替え装置を用い、複数のコンテキストを切り替える
コンテキスト切り替え方法であって、

コンテキストの切り替えが発生した場合、入力された切り替え命令と新た
10 に入れ替えるスレッド識別子(スレッドID)に基付き、前記コンテキストキャッ
シュに記憶されているスレッドのコンテキストを識別するためのスレッド識別
子(スレッドID)を記憶する前記スレッドIDテーブルを検索し、

新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッ
シュのアドレスと、実行中のコンテキストが記憶されている前記レジスタファイ
15 ルのレジスタ識別子を求め、

求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求
めたレジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジス
タファイル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテ
キストスイッチ専用バスを介して入れ替え、退避又は復帰を実行する
20 前記コンテキスト切り替え方法。

11. コンテキストを入れ替えるスワップ命令を実行する場合、オペレーティング
システム等のソフトウェアからスレッド識別子(スレッドID)をオペランドとして
含むスワップ命令が発行されると、前記レジスタファイルから前記コンテキ
25 ストキャッシュへ実行中のスレッドのコンテキストを退避すると同時に、前記コ
ンテキストキャッシュから前記レジスタファイルへ新しく入れ替えるスレッドの
コンテキストを送り、自動的に必要な数だけ前記レジスタファイル内のデー
タと前記コンテキストキャッシュ内のデータを入れ替えることを特徴とする請求
項10に記載のコンテキスト切り替え方法。

12. コンテキストを退避するバックアップ命令を実行する場合、オペレーティングシステム等のソフトウェアからスレッド識別子(スレッドID)をオペランドとして含むバックアップ命令が発行されると、前記レジスタファイルから前記コン
5 テキストキャッシュへのコンテキストのデータ転送を行い、前記コンテキストキャッシュから前記レジスタファイルへのコンテキストのデータ転送は行わないことを特徴とする請求項10に記載のコンテキスト切り替え方法。
13. コンテキストを復帰するリストア命令を処理する場合、オペレーティングシステム等のソフトウェアからスレッド識別子(スレッドID)をオペランドとして含む
10 リストア命令が発行されると、前記コンテキストキャッシュから前記レジスタファイルへのコンテキストのデータ転送を行い、前記レジスタファイルから前記コンテキストキャッシュへのコンテキストのデータ転送は行わないことを特徴とする請求項10に記載のコンテキスト切り替え方法。
14. 演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関する
15 コンテキストが記憶されたレジスタファイルと、
前記レジスタファイルに接続され、コンテキストをキャッシュするためのコンテキストキャッシュと、
20 前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキスト専用バスと
を備えたコンテキスト切り替え装置を用い、コンピュータが複数のコンテキストを切り替えるコンテキスト切り替えプログラムであって、
コンテキストの切り替えが発生した場合、入力された切り替え命令と新たに
25 に入れ替えるスレッド識別子(スレッドID)に基付き、前記コンテキストキャッシュに格納されているスレッドのコンテキストを識別するためのスレッド識別子(スレッドID)を記憶する前記スレッドIDテーブルを検索するステップと、
新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッシュのアドレスと、実行中のコンテキストが記憶されている前記レジスタファイ

ルのレジスタ識別子を求めるステップと、

求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求めたレジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジスタファイル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテキストスイッチ専用バスを介して入れ替え、退避又は復帰を実行するステップと

をコンピュータに実行させるためのコンテキスト切り替えプログラム。

15. 演算論理ユニット又はメモリアクセスユニットで実行すべきスレッドに関するコンテキストが記憶されたレジスタファイルと、

前記レジスタファイルに接続され、コンテキストをキャッシュするためのコンテキストキャッシュと、

前記レジスタファイルと前記コンテキストキャッシュを接続するコンテキストスイッチ専用バスと

- 15 を備えたコンテキスト切り替え装置を用い、コンピュータが複数のコンテキストを切り替えるコンテキスト切り替えプログラムを記録したコンピュータ読み取り可能な記録媒体であって、

コンテキストの切り替えが発生した場合、入力された切り替え命令と新たに入れ替えるスレッド識別子(スレッドID)に基付き、前記コンテキストキャッシュに格納されているスレッドのコンテキストを識別するためのスレッド識別子(スレッドID)を記憶する前記スレッドIDテーブルを検索するステップと、

新たに入れ替えるコンテキストが記憶されている前記コンテキストキャッシュのアドレスと、実行中のコンテキストが記憶されている前記レジスタファイルのレジスタ識別子を求めるステップと、

- 25 求めたアドレスに基付き前記コンテキストキャッシュをアクセスし、且つ、求めたレジスタ識別子に基付き前記レジスタファイルをアクセスし、前記レジスタファイル及び／又は前記コンテキストキャッシュのコンテキストを前記コンテキストスイッチ専用バスを介して入れ替え、退避又は復帰を実行するステップと

をコンピュータに実行させるためのコンテキスト切り替えプログラムを記録したコンピュータ読み取り可能な記録媒体。

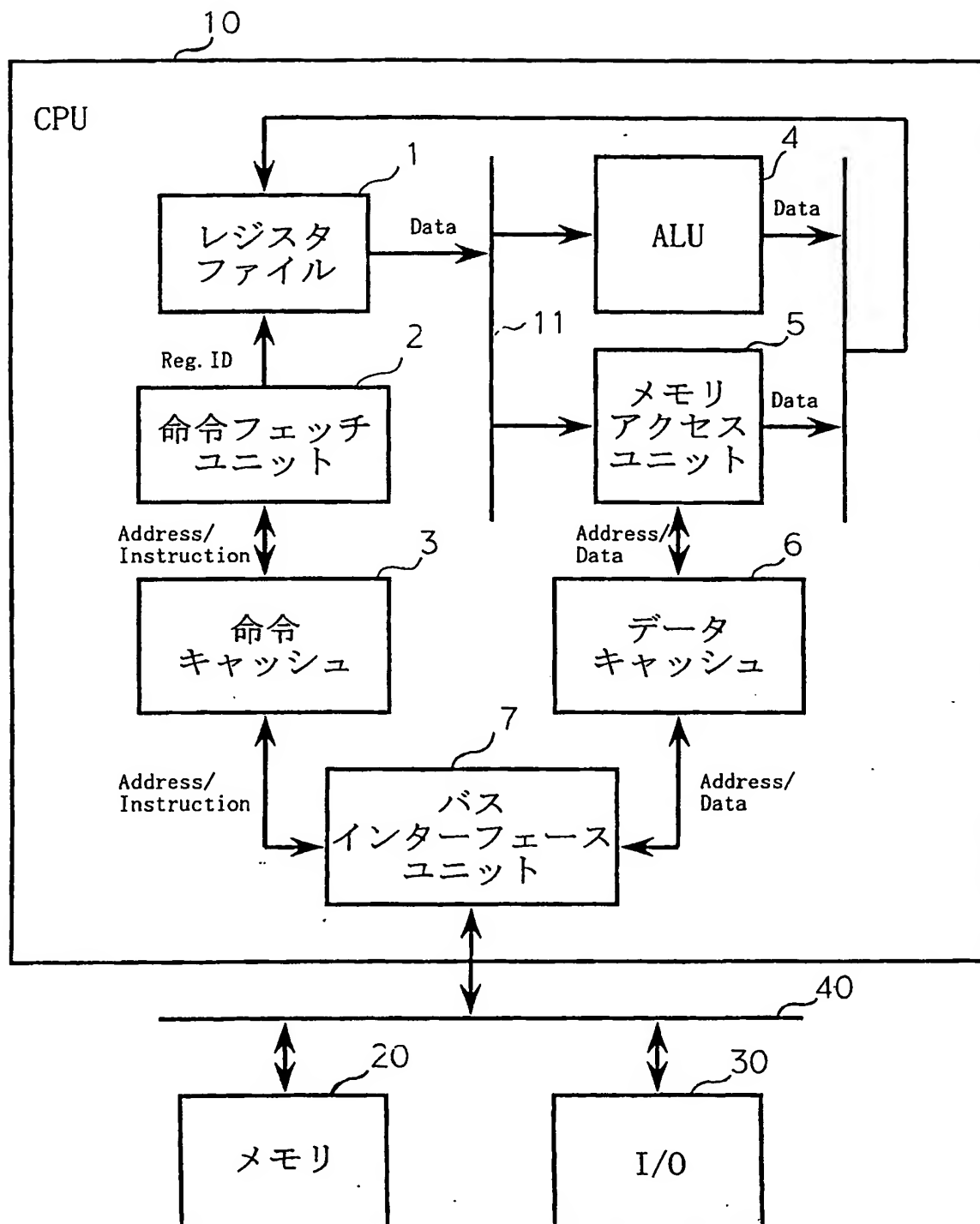


図 1

2 / 8

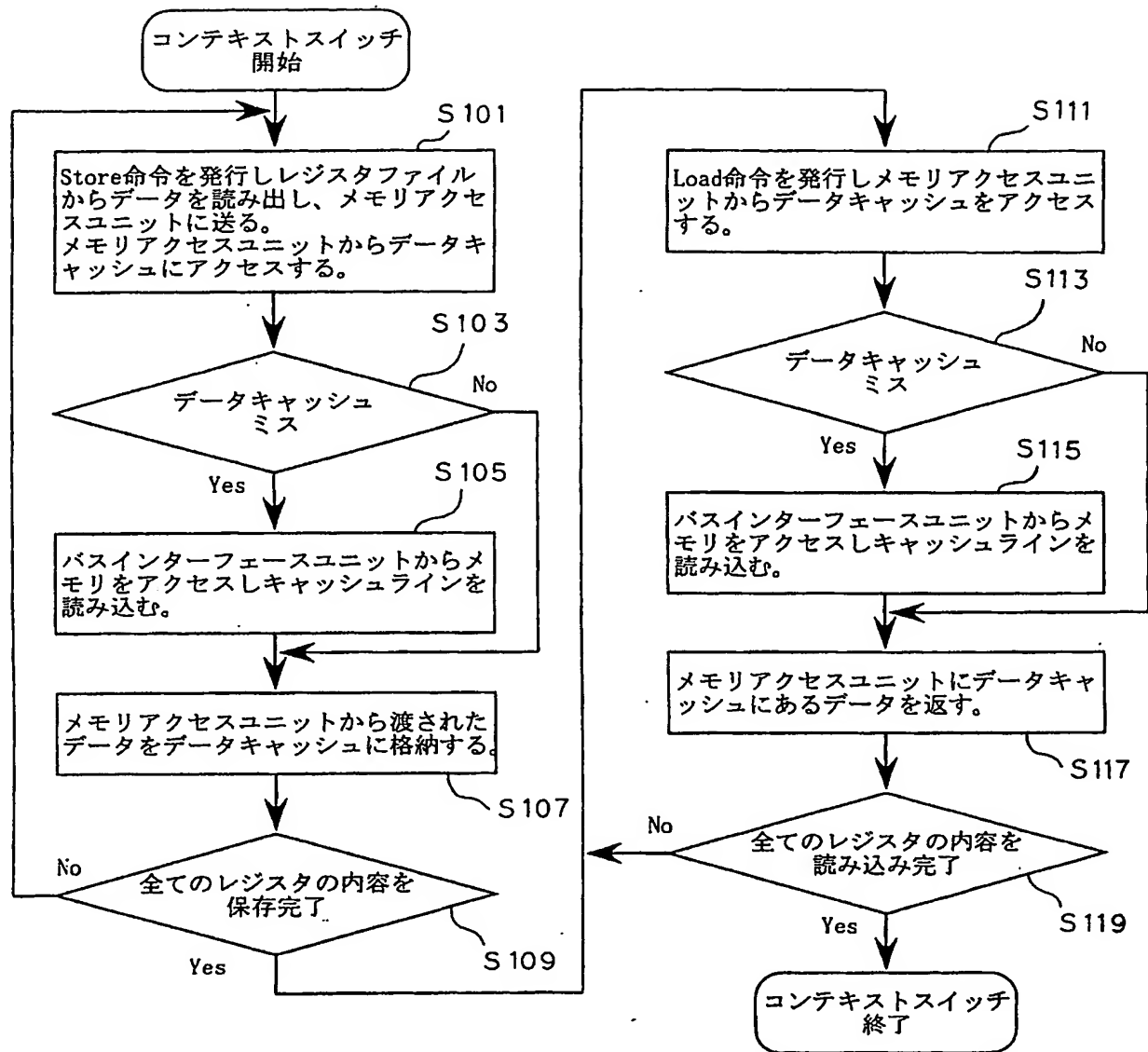


図 2

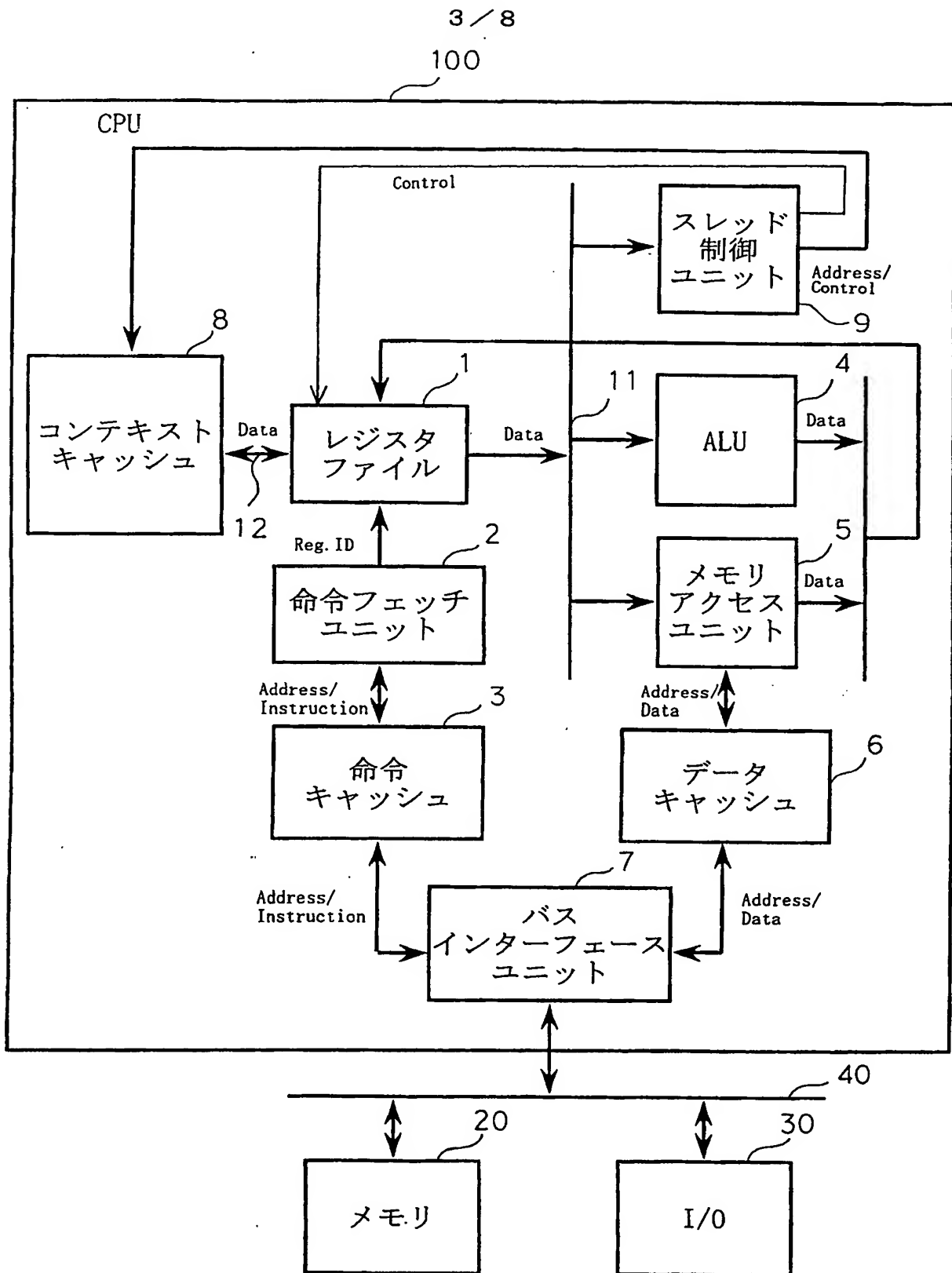


図 3

4 / 8

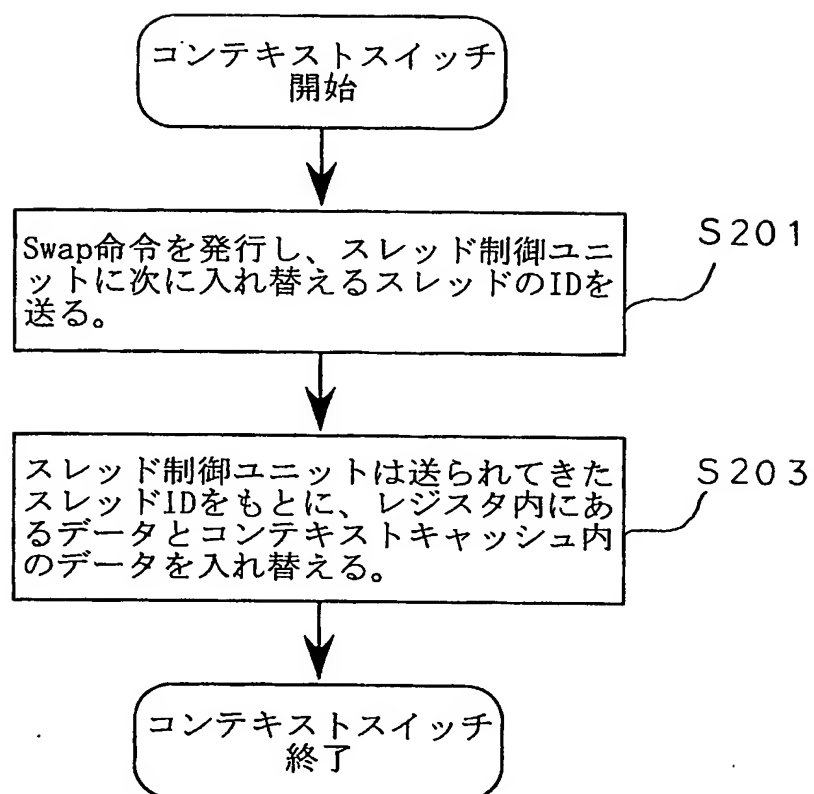


図 4

5 / 8

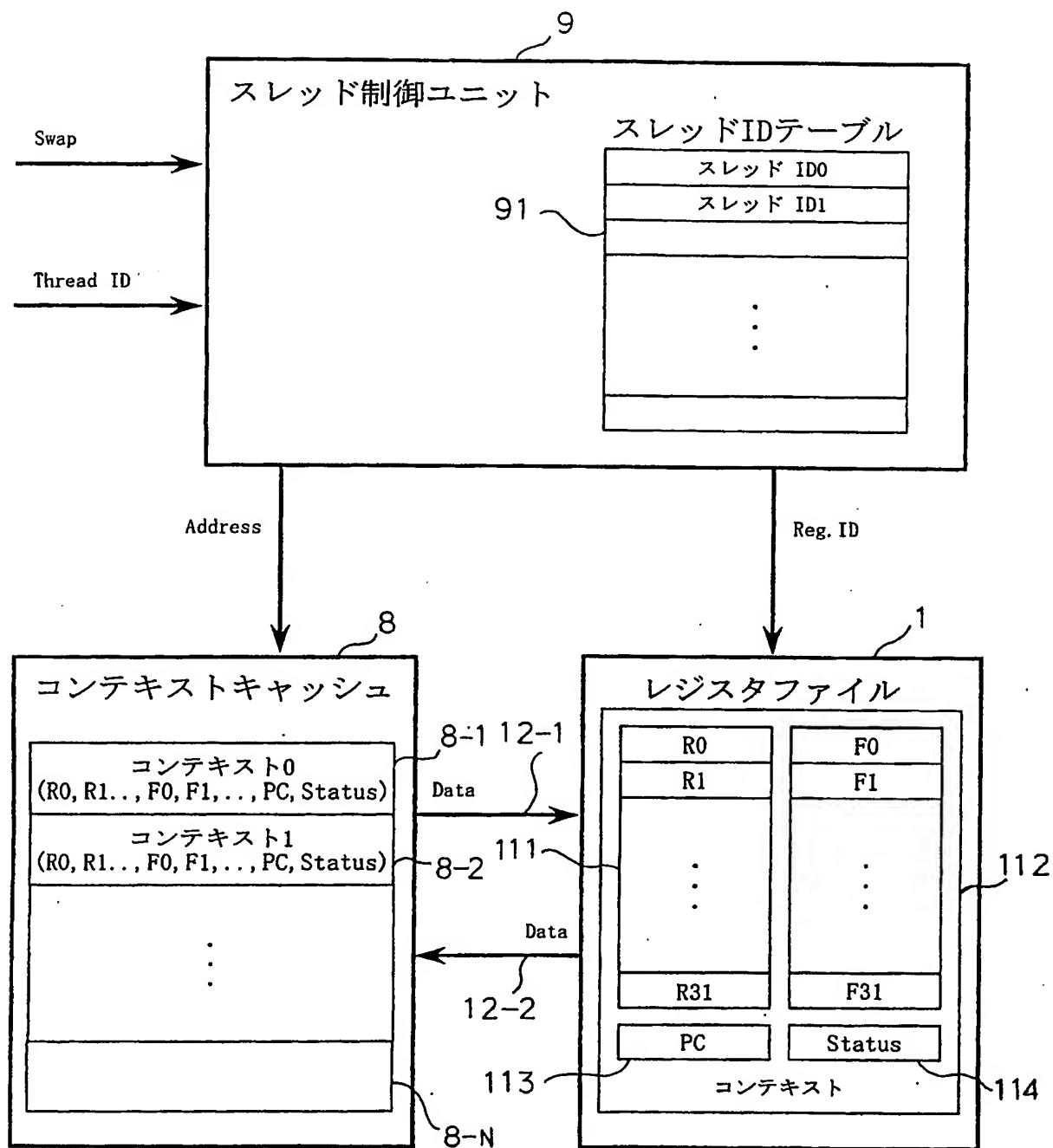


図 5

6 / 8

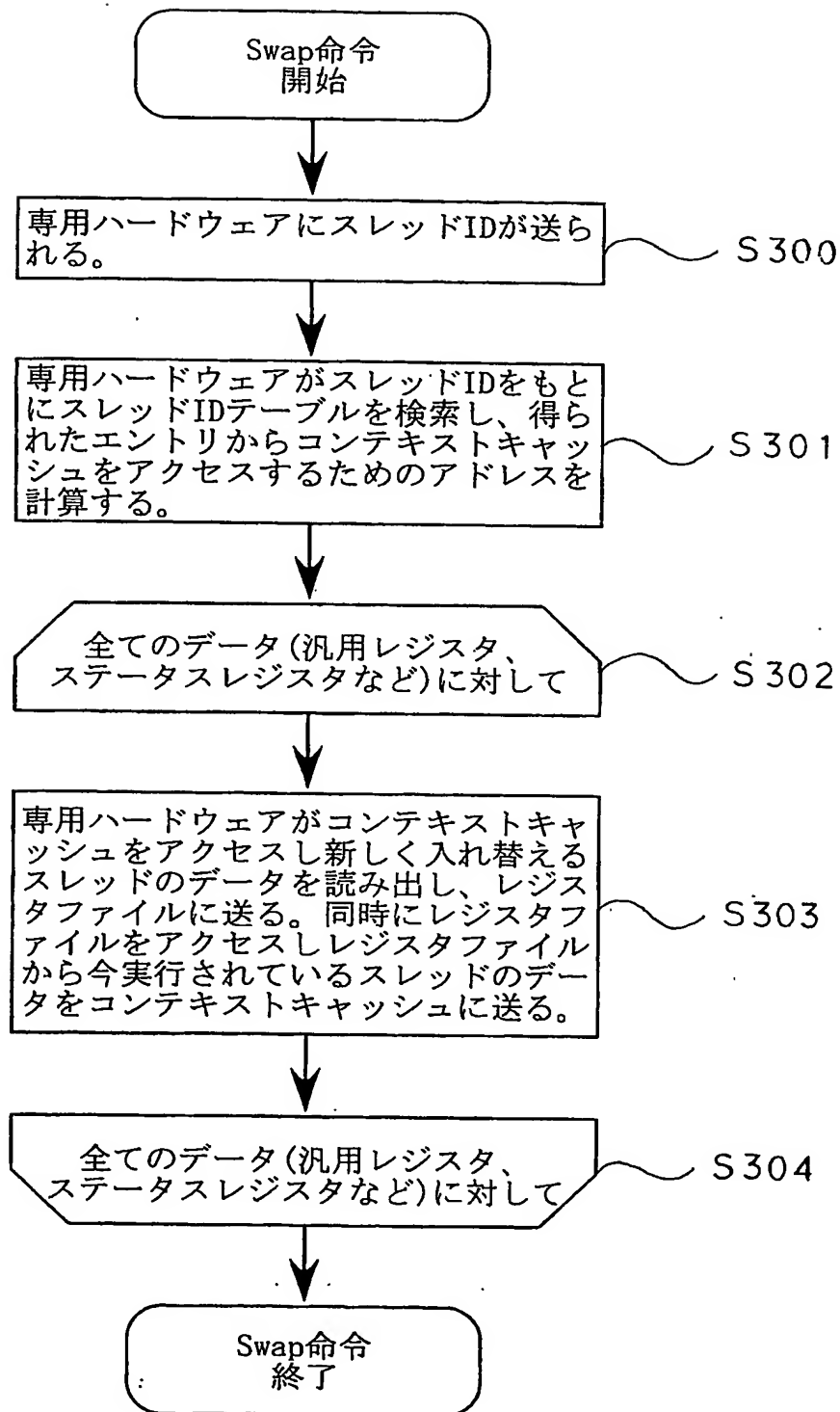


図 6

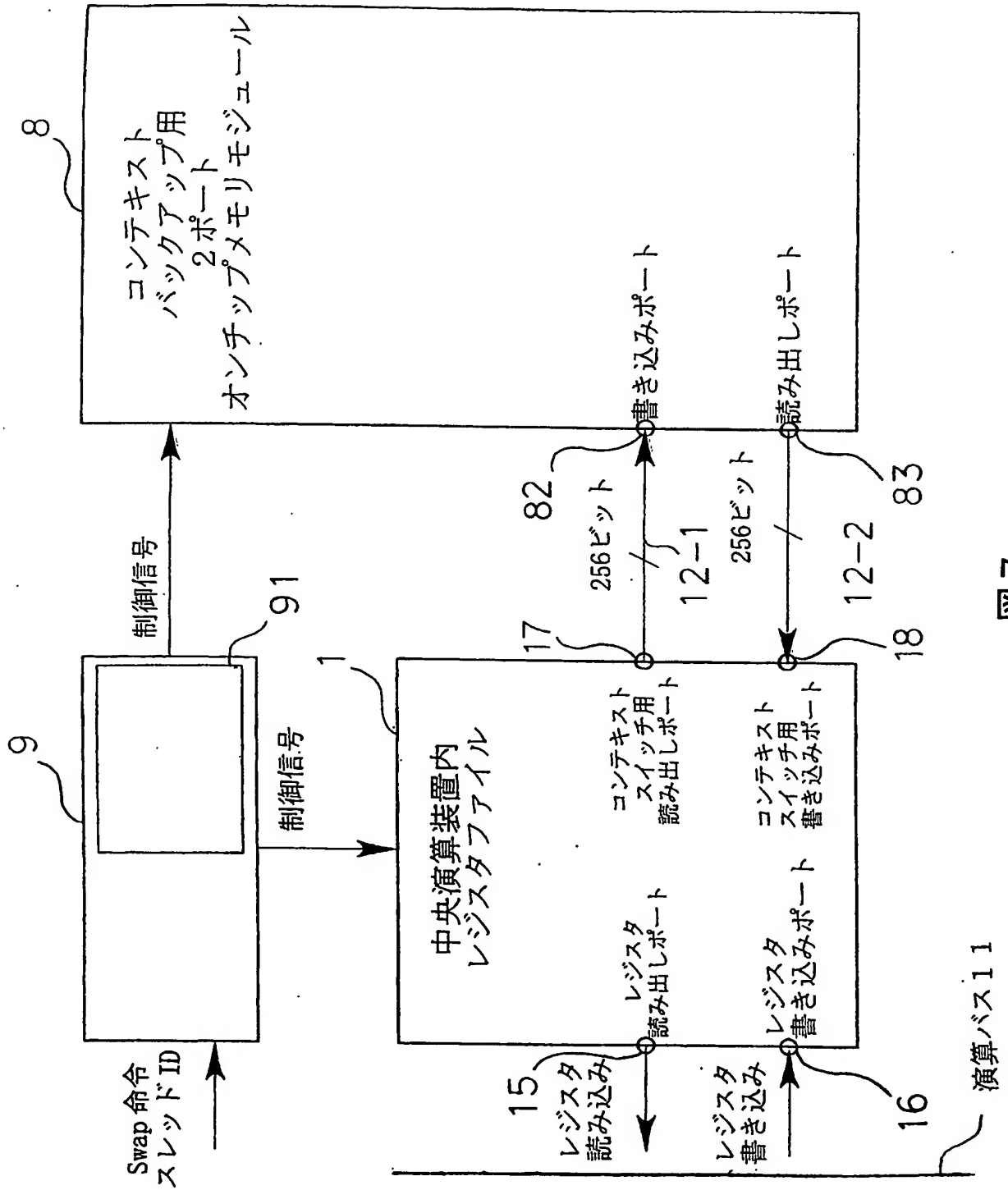


図7

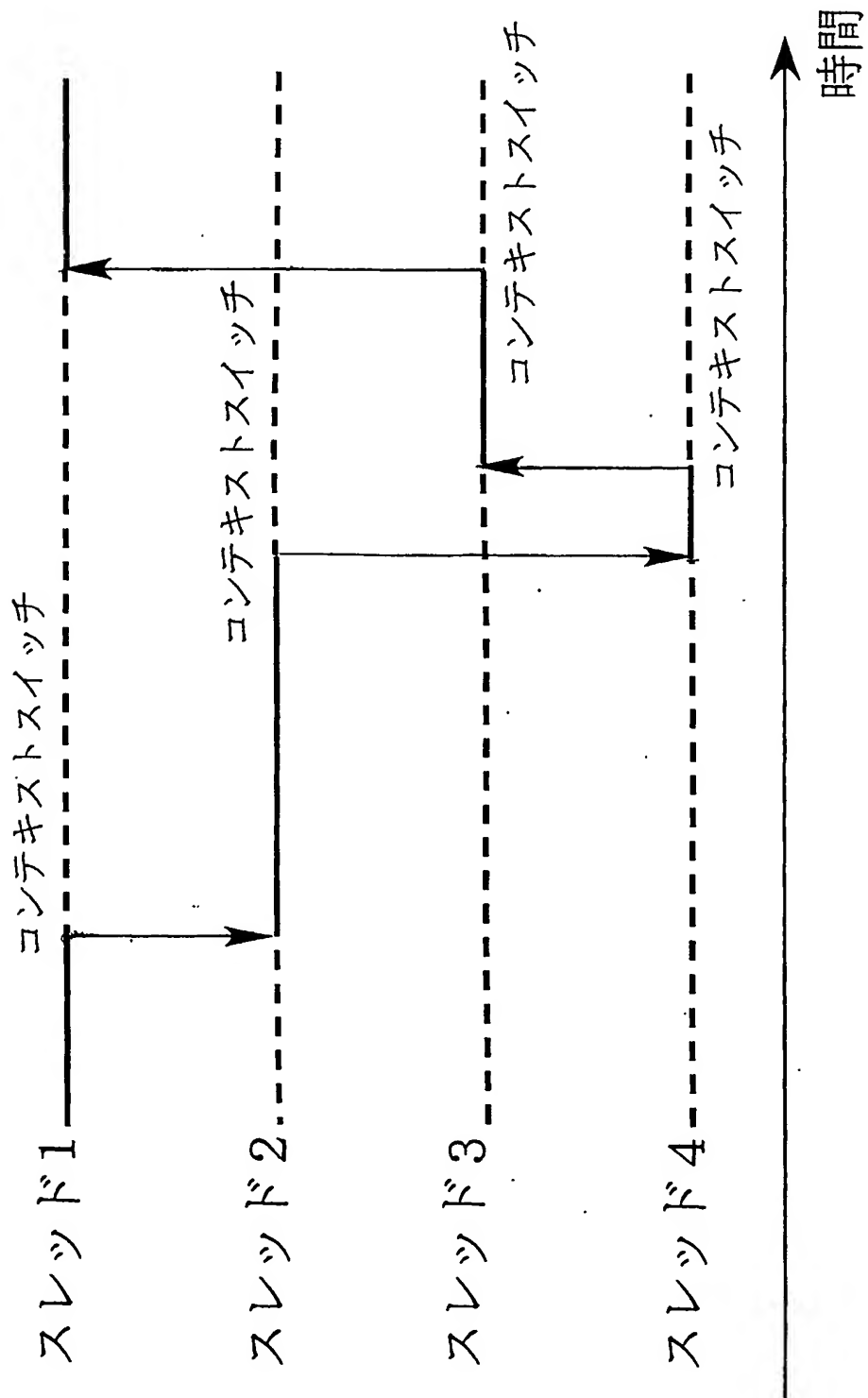


図8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP03/15838

A. CLASSIFICATION OF SUBJECT MATTER

Int.Cl⁷ G06F9/46

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Int.Cl⁷ G06F9/46-9/54

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2004
Kokai Jitsuyo Shinan Koho	1971-2004	Toroku Jitsuyo Shinan Koho	1994-2004

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	Guangzuo, C. et al., Parallel Replacement Mechanism for MultiThread, Advances in Parallel and Distributed Computing 1997 Proceedings, 1997, pages 338 to 334; particularly, pages 340,	1-15
Y	Wills, D.S. et al., Pica: An Ultra-Light Processor for High-Throughput Application, Computer Design: VLSI in Computers and Processors, 1993. ICCD '93. Proceedings, 1993, pages 410 to 414; particularly, pages 411 to 412	1-15
A	Kwak, H. et al., Effects of multithreading on cache performance, Computers, IEEE Transactions on, 1999, pages 176 to 184	1-15

☒ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance
 "E" earlier document but published on or after the international filing date
 "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
 "O" document referring to an oral disclosure, use, exhibition or other means
 "P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
 "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
 "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
 "&" document member of the same patent family

Date of the actual completion of the international search
 23 January, 2004 (23.01.04)

Date of mailing of the international search report
 03 February, 2004 (03.02.04)

Name and mailing address of the ISA/
 Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP03/15838

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2000-242505 A (Mitsubishi Electric Corp.), 08 September, 2000 (08.09.00), Par. Nos. [0044] to [0047] (Family: none)	1-15
Y	JP 3-9431 A (NEC Corp.), 17 January, 1991 (17.01.91), Full text (Family: none)	12-13
A	JP 10-11301 A (Seiji IMAI), 16 January, 1998 (16.01.98), Full text (Family: none)	1-15
A	JP 2000-76079 A (NEC Corp.), 14 March, 2000 (14.03.00), Full text (Family: none)	1-15
A	JP 2002-513182 A (Infineon Technologies North America Corp.), 08 May, 2002 (08.05.02), Full text & WO 99/56209 A1 & EP 1073956 A1 & US 6378065 B1	1-15

A. 発明の属する分野の分類 (国際特許分類 (IPC))

Int. Cl⁷ G06F 9/46

B. 調査を行った分野

調査を行った最小限資料 (国際特許分類 (IPC))

Int. Cl⁷ G06F 9/46 - 9/54

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報 1922-1996年
 日本国公開実用新案公報 1971-2004年
 日本国実用新案登録公報 1996-2004年
 日本国登録実用新案公報 1994-2004年

国際調査で使用した電子データベース (データベースの名称、調査に使用した用語)

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
Y	Guangzuo, C. et al., Parallel Replacement Mechanism for MultiThread, Advances in Parallel and Distributed Computing, 1997 Proceedings, 1997, pp. 338-334, 特に、p. 340の記載	1-15
Y	Wills, D.S. et al., Pica: An Ultra-Light Processor for High-Throughput Application, Computer Design:VLSI in Computers and Processors, 1993. ICCD '93. Proceedings, 1993, pp. 410-414, 特に、pp. 411-412	1-15

☒ C欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」 特に関連のある文献ではなく、一般的技術水準を示すもの
 「E」 国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの
 「L」 優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献 (理由を付す)
 「O」 口頭による開示、使用、展示等に言及する文献
 「P」 国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」 国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」 特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」 特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」 同一パテントファミリー文献

国際調査を完了した日

23.01.04

国際調査報告の発送日

03.2.2004

国際調査機関の名称及びあて先

日本国特許庁 (ISA/JP)

郵便番号100-8915

東京都千代田区霞が関三丁目4番3号

特許庁審査官 (権限のある職員)

殿川 雅也

5B

9646

電話番号 03-3581-1101 内線 3585

C (続き). 関連すると認められる文献		
引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求の範囲の番号
A	Kwak, H. et al., Effects of multithreading on cache performance, Computers, IEEE Transactions on, 1999, pp. 176-184	1-15
A	JP 2000-242505 A (三菱電機株式会社) 2000.09.08, 段落番号0044-0047 (ファミリーなし)	1-15
Y	JP 3-9431 A (日本電気株式会社) 1991.01.17, 全文 (ファミリーなし)	12-13
A	JP 10-11301 A (今井正治) 1998.01.16, 全文 (ファミリーなし)	1-15
A	JP 2000-76079 A (日本電気株式会社) 2000.03.14, 全文 (ファミリーなし)	1-15
A	JP 2002-513182 A (インフィニオン テクノロジーズ ノース アメ리카 コーポレーション) 2002.05.08, 全文 & WO 99/56209 A1 & EP 1073956 A1 & US, 6378065 B1	1-15